(1) Evaluation and Cross Validation (6 Pages) + (2 pages)

- Evaluate the performance of algorithm

Different metrics can be used - 1) Error metric
2) Accuracy
3) Precision and Recall
} Test on the Sample.

- Train on the training set and test on the testing set.

- Evaluating the performance of learning systems is important because:
→ Learning systems are usually designed to predict the class of "future" unlabelled data points.

- Typical choices for performance evaluations: i) Error
ii) Accuracy
iii) Recall / precision

- Typical choices for sampling method - i) Test / Train set
ii) K fold cross validations.

Evaluating predictions -

- Suppose we want to make a prediction of value for target feature on example x:
- y is the observed value of target feature on example x.
- $\hat{y}$ is the predicted value of target feature on example x. i.e, $\hat{y} = h(x)$
- How is the error measured?
if $y = \hat{y}$ then no error.
$y \neq \hat{y}$ then there is error. (Reducible error)

Error = Reducible error + Irreducible error
RE = Bias^2 + Variance.

Type of error - i) Absolute error → $h(x) - y$ (on single training example)
$= \frac{1}{n} \Sigma |h(x) - y|$ (on n training example take average).

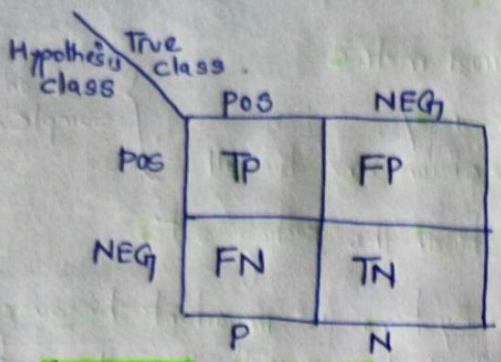
H → Given hypothesis space.
S → Given training example
h → Training algo, gives h belonging to H.

ii) Sum of square error.
$\frac{1}{n} \sum_{i=1}^{n} (h(x) - y)^2$

Note - Absolute error and SSE are mainly used for regression.

iii) For classification problem, check misclassification.
No of misclassification = $\frac{1}{n} \sum_{i=1}^{n} \delta(h(x), y)$
δ will return 1, if $h(x)$ and y different / 0, if they are same

iv) Confusion matrix



Two type of mistake
Model is wrongly classifying the algo class.

i) FP
ii) FN

Note - For 3 class, we will have 3 x 3 confusion matrix.

$$\text{Accuracy} = \frac{TP + TN}{P + N}, \quad P = TP + FN, \quad N = FP + TN.$$

Precision = Out of example, the learning algorithm mark as positive how many are correctly positive.

$$= \frac{TP}{TP + FP}$$

Recall = How many positive example does the learning algo retrives as positive.

$$= \frac{TP}{TP + FN} = \frac{TP}{P}.$$

Sample error and True error

Sample error - Sample error of hypothesis f with respect to target function c and data sample S is :

$$error_S(f) = \frac{1}{n} \sum_{x \in S} \delta(f(x), c(x)) = \text{avg(misclassfn)}$$

True error - Denoted $error_D(f)$ of hypothesis is f with respect to target function C and distribution D, is the probability that h will misclassify an instance drawn at random according to D.

$$error_D(f) = Pr_{x \in D}[f(x) \neq c(x)]$$

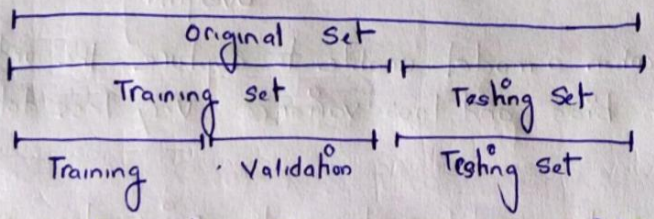
- Error we get in sample is called sample error and actual error is called true error.
- The error may come because given the hypothesis space the search algorithm is not exhaustively searching the hypothesis space, but making certain simplification is called search bias.

③

- Variance error - Error may be due to the limited size of the sample that we use for testing then it is called variance error.
  - This may rise because feature which we are using or language (vocabulary) that we are using is not sufficient to capture everything about the task. (It is also known as noise).
- So sample error can be different from true error. We have to find those error. That is why we use training set and test set.
- If test set is small, accuracy may be high.
- Training set is small, variance error may be high. Overfitting may be also come.

Cross validation -

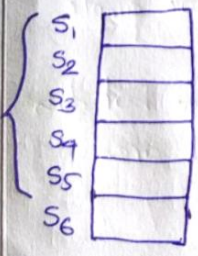
- Neither training set can be small nor test set.
- Use cross validation for that.



- Validation set is used during tuning during training to tune the model parameters.

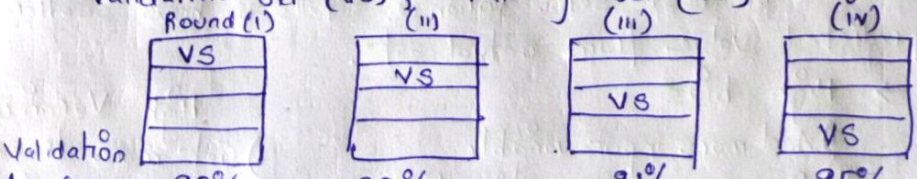
K- ~~fold~~ fold cross validation - To cover all training dataset & Validate

1. Split the data into K equal parts.
2. Perform K rounds of learning, on each round
   a) 1/K of data is held out as a test set
   b) Remaining example are used as training data.
3. Compute the average test set score of K rounds.



Round i - Use S_i for testing
$S - S_i$ use for training

Validation Set (VS) Training Set (TS)

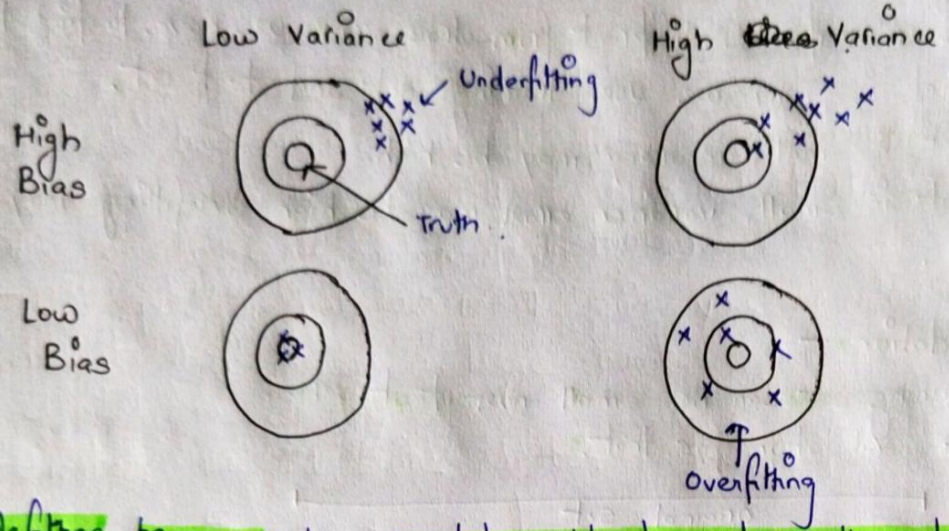


Validation Accuracy - 93% 90% 91% 95%

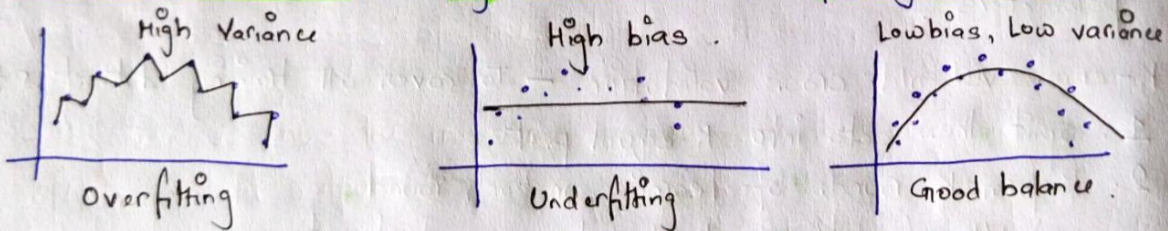
Final accuracy = Avg (All accuracy)

Trade-off -

- In ML, there is a trade-off between.
  - Complex hypothesis that fit training data well.
  - Simpler hypothesis that may generalised well.
- As the amount of training data increases, the generalization error decreases.



- Underfitting happens when a model unable to capture the underlying pattern of data. Have high bias and low variance. Very less data to build accurate model.
- Overfitting happens when model captures the noise along with data. Model have low bias and high variance. It often poor genalizability.



- Bias is how far are predicted value and actual value. If average predicted value are far from actual value then it is high bias. When a model is high bias then it implies that model is too simple and does not capture complexity. Thus underfitting.
- Variance occurs when model performs good on training dataset but does not do well on test set. Variance tells us how scattered are the predicted value from actual set.

Solution → High bias

i) Add more input variable
ii) Decrease regularization term
iii) Add more complexity

High Variance

i) Get more training data.
ii) Reduce input feature.
iii) Increase regularization term.

(5)

- Error = Reducible error + Irreducible error.

Reducible error = Bias² + Variance

- Irreducible error can be reduced, no matter what algo we used.

Confusion Matrix In Depth -

i) Recall ii) Precision iii) Specificity
iii) Accuracy v) AUC-ROC curve.

Actual Value.

Positive (1) Negative (0).

Predicted Value

	Positive (1)	Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

i) TP ii) FP iii) FN iv) TN.

i) True positive →
Interpretation - You predicted positive and it's true.
Model predicted women is pregnant and she actually is.

ii) True Negative →
Interpretation - You predicted negative and it's true.
Model predicted that a man is not pregnant and he actually is not

Actual Value.

Predicted value		
	TP Female → pregnant	FP Male → Pregnant (Type 1 error)
	FN → Female → Not pregnant (Type 2 error)	TN. Male → Not pregnant

iii) False positive - (Type 1 error)
- Interpretation → You predicted positive and it's false.
- You predicted that man is pregnant but he actually is not.

iv) False negative - (Type 2 error)
- Interpretation → You predicted negative and it's false.
- You predicted that a woman is not pregnant but she actually is.

Example →

(Actual) y	y-pred	(Pred) output
0	0.5	0 - TP -
1	0.9	1 - TN - Pred
0	0.7	1 - FN -
1	0.7	1 - TN -
1	0.3	0 - FP -
0	0.4	0 - TP -
1	0.5	0 - FP -

Actual

Predicted	0	1
0	TP 2	FP 2
1	FN 1	TN 2

Recall = $\frac{TP}{TP+FN} = \frac{2}{3}$.

Precision = $\frac{TP}{TP+FP} = \frac{2}{4} = \frac{1}{2}$.

Accuracy = $\frac{TP+TN}{All} = \frac{2+2}{2+2+2+1} = \frac{4}{7}$.

Recall - Out of all positive class, how many we predicted correctly. It should be high as possible.

Precision - Out of all positive class, we have predicted correctly, how many actually cor

F-measure → It is difficult to compare two models with low precision and high recall or vice versa.
So use F-score. F score helps to measure Recall and precision.

$$F\text{-measure} = \frac{2 * Recall * Precision}{Recall + precision}$$

F1 Score → F1 score is used to measure a test's accuracy.
- F1 score is the harmonic mean between precision and recall.
- The range of F1 score is [0,1].
- It tells how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it doesn't miss a significant number of instances).
- High precision but low recall, gives you an extremely accurate, but it then misses a large number of instances that are difficult to classify.
- Greater the F1 score, better is the performance of model.

$$F_1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{Recall}}$$

F1 score try to find the balance between precision and recall.

Mean Absolute Error →
- It is the average of the difference between original and predicted values.
- It gives us the measure of how far the predictions were from actual output.
- However, they don't give us any idea of the direction of the error i.e, whether we are under predicting or over predicting the data.

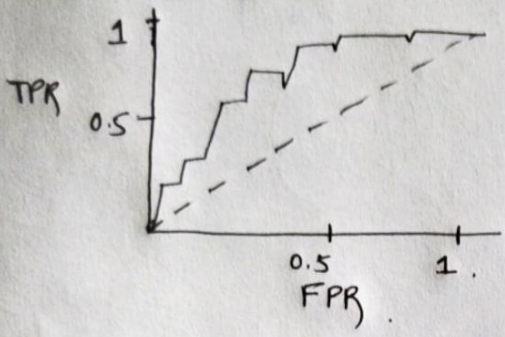
$$\text{Mean Absolute error (MAE)} = \frac{1}{N} \sum_{j=1}^{N} |y_j - \hat{y}_j|$$

Mean Square Error →
- MSE is quite similar to MAE, the only difference being that MSE takes the average of square of difference between original values and predicted values.
- As we take square of error, the effect of larger errors become more pronounced than smaller errors, hence model can now focus on large errors.

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

(7) Area Under Curve →

- AUC of a classifier is equal to the probability that the classifier will rank a randomly choosen positive example higher than a randomly choosen negative example.
- It is a graph between False positive and True positive. AUC is the area under the curve of plot FPR vs TPR at different point of [0,1].



- Higher the value of ROC, better is the performance of model.

Metrics to evaluate ML algorithm —

1) Accuracy 2) Confusion matrix 3) Area Under Curve 4) F1 score.
5) Mean Absolute error 6) Mean square error.